

COMPUTING METHODS IN OPTIMIZATION PROBLEMS

AFOSR 65-0603

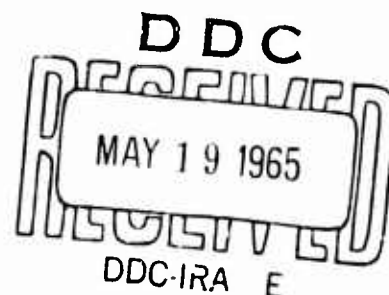
SEVERAL TRAJECTORY OPTIMIZATION TECHNIQUES

Part II: Application

H. Gardner Moyer

Gordon Pinkham

Research Department  
Grumman Aircraft Engineering Corporation  
Bethpage, New York



Abstract

This paper discusses the application of the three optimization methods — steepest descent, second variation, and generalized Newton-Raphson — to the problem of minimum time, low thrust, circle-to-circle transfer. Details of computational techniques that have proved successful in practice are presented. The number of iteration cycles and the time used by the computer are given for each method.

Introduction and Statement of Problem

This paper describes the application of the three basic optimization schemes covered in Part I: Discussion to a specific problem. This problem is that of minimizing the transfer time of a low-thrust ion rocket between the orbits of Earth and Mars. Applications of the gradient method and of the second variation method to this problem have been reported in (1), (2), and (4), and therefore only a brief sketch of interesting variations in these computational procedures and of the results will be included here.

To simplify the problem as much as possible the rocket's thrust level was assumed constant, and thus the single control variable is the thrust direction. Further, the orbits of Earth and Mars were assumed to be circular and coplanar, and the gravitational attractions of the two planets on the vehicle were neglected. The following system parameters for the low-thrust vehicle were adopted

15-P

COPY	1	OF	1	90
HARD COPY	\$ . 1.00			
MICROFICHE	\$ . 0.50			

REPRODUCTION IN WHOLE OR  
IN PART IS PERMITTED FOR ANY  
PURPOSE OF THE U.S. GOVERNMENT

GRD-119 II

PROCESSING COPY  
ARCHIVE COPY

AD615153

## COMPUTING METHODS IN OPTIMIZATION PROBLEMS

from (1).

Initial Mass, $m_0$	46.58 slugs
Specific Impulse	5700 sec
Propellant Consumption, $\dot{m}$	$-6.937 \times 10^{-7}$ slugs/sec
Thrust, $T$	0.127 lb
Thrust/Initial Weight	$0.9 \times 10^{-4}$

The equations of motion were given by:

### Radial Velocity

$$\dot{r} = f^{(1)} = u \quad (1)$$

### Radial Acceleration

$$\dot{u} = f^{(2)} = \frac{v^2}{r} - \frac{\mu}{r^2} + \frac{T \sin \alpha}{m_0 + \dot{m}t} \quad (2)$$

### Circumferential Acceleration

$$\dot{v} = f^{(3)} = -\frac{uv}{r} + \frac{T \cos \alpha}{m_0 + \dot{m}t} \quad (3)$$

where  $u$  and  $v$  are the radial and circumferential velocities respectively;  $r$  is the radius; and  $\alpha$  is the thrust direction angle measured from the local horizontal. All the initial and final values of the state variables were specified, and the quantity to be minimized was  $t_f$ , the final time.

## The Three Methods as Applied to the Sample Problem

### Gradient Method

As stated in Part I, the gradient methods utilize a penalty function which for this problem becomes

$$P' = t + \frac{1}{2} \sum_{i=1}^3 k_i (x_{i_f} - \tilde{x}_i)^2 \quad (4)$$

## COMPUTING METHODS IN OPTIMIZATION PROBLEMS

As was also stated in Part I, the criterion for the termination of a trajectory is

$$\dot{P}' = 1 + \sum_{i=1}^3 k_i (x_{i_f} - \tilde{x}_i) f_f^{(i)} = 0 \quad (5)$$

Utilizing

$$\lambda_{i_f} = \frac{\partial P'}{\partial x_i} = k_i (x_{i_f} - \tilde{x}_i) \quad (6)$$

we have

$$1 + \sum_{i=1}^3 \lambda_{i_f} f_f^{(i)} = 0 \quad (7)$$

This equation scales the multipliers.

Since by the "multiplier rule" all the final  $\lambda$ 's cannot vanish, the extremal that minimizes  $P'$  must deviate from the desired values  $\tilde{x}_i$ . We will present two methods of reducing the deviation to within tolerable limits.

We will assume that the errors are small enough so that the extremal that minimizes  $P'$  is close to the extremal that passes through the desired end-point. By close we mean that the two extremals have approximately the same final  $\lambda$ 's. If we increase the  $k$ 's and re-converge, this new extremal will also have approximately the same final  $\lambda$ 's. It then follows from Eq. (6) that the deviations will be reduced in proportion to the increase in the  $k$ 's.

Thus the first method of reducing the errors is to "build up the  $k$ 's." This method was often unsuccessful however, due to inexact interpolation for the final values and other numerical errors that prevented convergence. Therefore, the method of "shifting boundary values" was devised and has proved invariably successful.

After convergence, the boundary values are redefined by subtracting  $\Delta x_i = x_{i_f} - \tilde{x}_i$  from  $\tilde{x}_i$ . We then apply the gradient process to obtain the minimum of

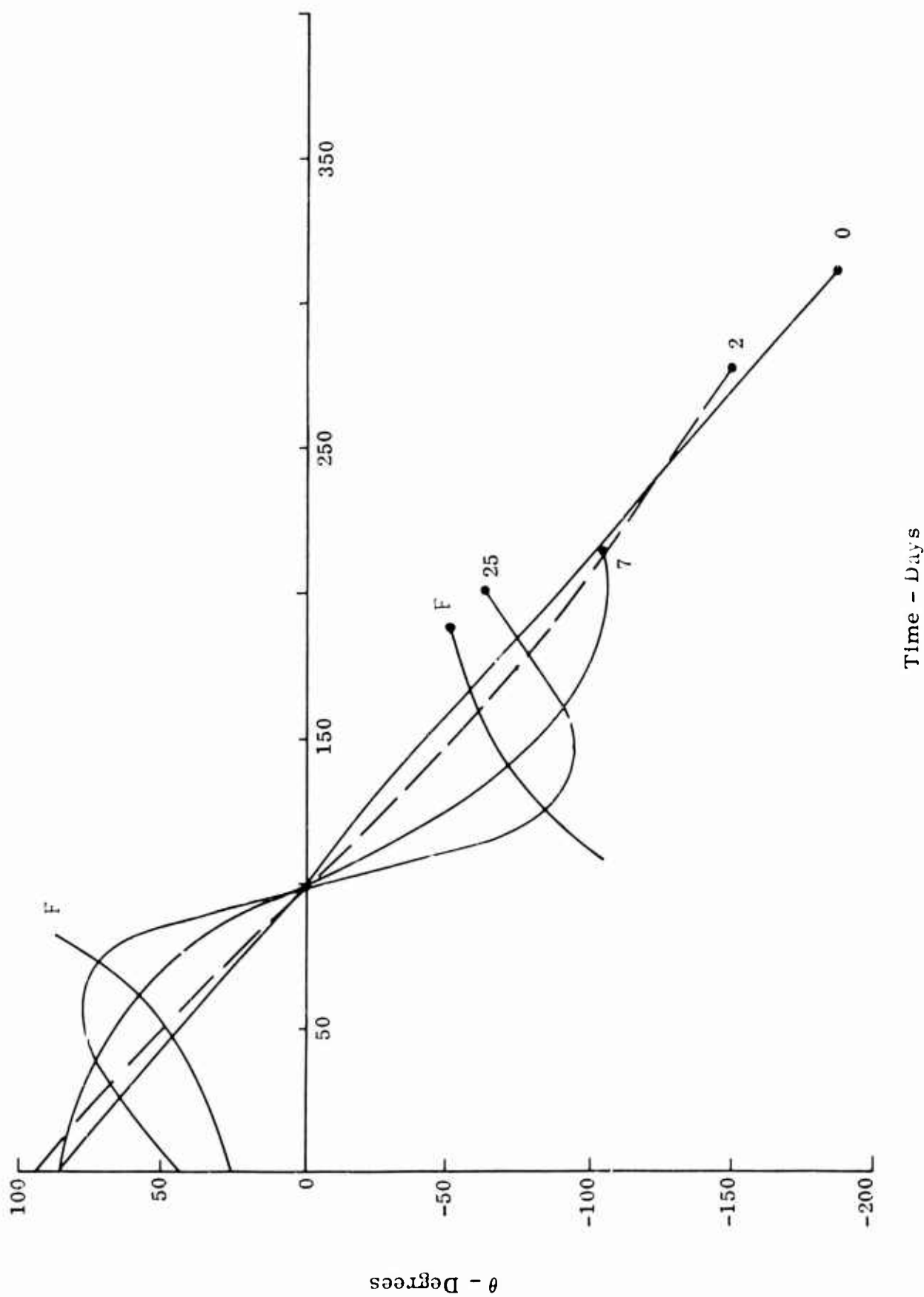


Fig. 1 Control Angle Programs From Gradient Technique

$$P' = t + \sum_{i=1}^3 k_i (x_{i_f} - \tilde{x}_i + \Delta x_i)^2 \quad (8)$$

If the final  $\lambda$ 's were unchanged, the extremal would pass directly through the desired end-point. In practice, several boundary shifts may be necessary.

Applying a gradient scheme incorporating the boundary shift technique to the above problem of minimizing the time of low-thrust Earth to Mars trajectories, it was found that attainment of the minimum  $t$  of 193 days with convergence to within 0.1% of the boundary values was possible in 5 shifts with an average of 20 descents per shift. As programmed, a descent was a one dimensional search for a minimum of  $P'$  versus control variations calculated from the adjoint equations, and it required the computation of one adjoint solution and at most four trajectories. Figure 1 shows the initial, some intermediate, and the final  $\theta$  time-histories preceding the first boundary shift. As indicated in the figure, the first boundary shift took place after 25 descents. The converged curve, labeled F, is also included.

### Second Variation Method

As explained in Part I, the first stage of the second variation method also depends on a penalty function,  $P'$ . However, in this case  $P'$  is expanded to second order terms, and among these are terms in  $\delta t_f$ , the change in the final time. Because these terms greatly complicate an already lengthy numerical calculation,  $\delta t_f$  was set equal to zero in the expansion of  $P'$  when the second variation method was coded for the computer. This implies that the control variable increment  $\delta \theta(t)$  was then calculated to obtain the maximum reduction in  $P'$  at the nominal final time  $\bar{t}_f$  instead of  $\bar{t}_f + \delta t_f$ . Of course when the trajectory with control variable  $\theta + \delta \theta$  was computed, it was not terminated at time  $\bar{t}_f$  but rather at that point of the trajectory with minimum  $P'$ , as is done in the gradient programs. With respect to over-all computational time this technique represents a compromise between a true second variation calculation and additional programming complexity. For this particular problem it was advantageous to treat the problem as a

fixed time problem when computing  $\delta\theta$  using penalty functions.

The initial  $\theta(t)$  function in this case corresponds to constant circumferential thrust. This resulted in terminal boundary value errors that averaged 20 per cent. After 6 descent cycles, using the penalty function procedure, the terminal errors averaged 3 per cent with the transfer time at 180 days. After 5 additional cycles of the refinement process, the average boundary value error was reduced to 0.05 per cent and the transfer time had reached its minimum of 193 days. The over-all IBM 7094 computer time was 2 minutes, representing half the computer time required by the first order gradient program. Intermediate  $\theta(t)$  curves (numbered by descent) are shown in Fig. 2. Those labeled 6 or less come from the penalty function stage. The F-4 curve typifies the refinement stage. The converged curve is labeled F.

Experiments were made with fewer descent cycles in the penalty function stage. These cases all required more computer time indicating that it is best to be close to the desired extremal before going over to the refinement stage.

### Generalized Newton-Raphson Method<sup>1</sup>

The last of the three methods discussed in Part I is based on an algorithm for solving the two-point boundary value problem associated with the Euler-Lagrange equations. As outlined in (3), it required a fixed final time. Since the sample problem of this paper is a minimum time problem, the procedure was altered to suit this case. What follows is a brief description of the modified procedure and a discussion of the numerical results obtained by application of the method to the sample orbit transfer problem.

The two-point boundary value problem resulting from the Euler-Lagrange equations is given by

---

<sup>1</sup>This algorithm was apparently first suggested for boundary value problems by Hestenes (5), and was developed further by Kalaba (6). A convergence proof for N dimensional systems was given by McGill and Kenneth (3).

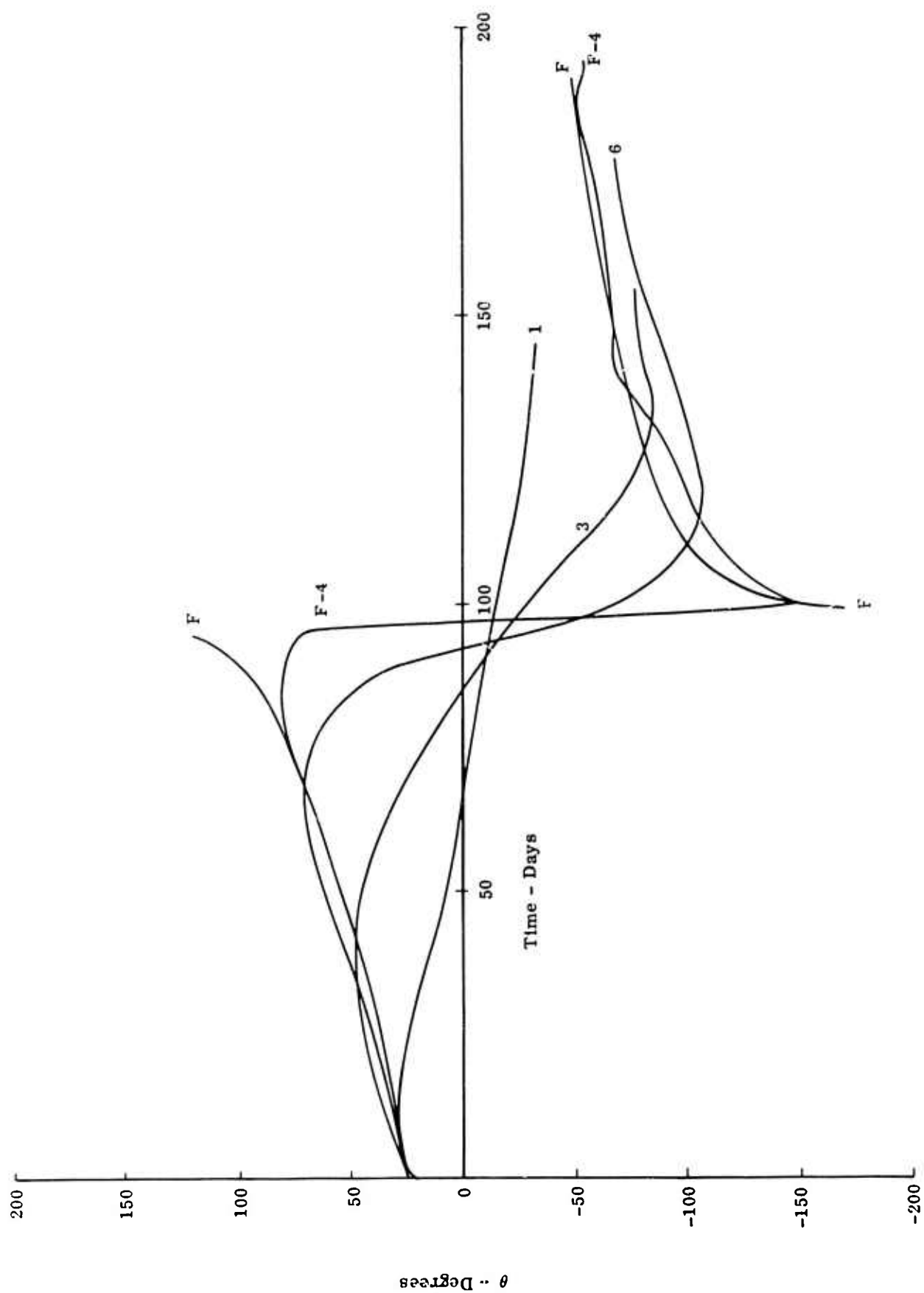


Fig. 2 Control Angle Programs From Second Variation Technique

# COMPUTING METHODS IN OPTIMIZATION PROBLEMS

$$\dot{r} = u = f^{(1)} \quad (9)$$

$$\dot{u} = \frac{v^2}{r} - \frac{\mu}{r^2} + a(t) \frac{\lambda_u}{(\lambda_u^2 + \lambda_v^2)^{\frac{1}{2}}} = f^{(2)} \quad (10)$$

$$\dot{v} = -\frac{uv}{r} + a(t) \frac{\lambda_v}{(\lambda_u^2 + \lambda_v^2)^{\frac{1}{2}}} = f^{(3)} \quad (11)$$

$$\dot{\lambda}_r = \left( \frac{v^2}{r^2} - 2 \frac{\mu}{r^3} \right) \lambda_u - \frac{uv}{r^2} \lambda_v = f^{(4)} \quad (12)$$

$$\dot{\lambda}_u = -\lambda_r + \frac{v}{r} \lambda_v = f^{(5)} \quad (13)$$

$$\dot{\lambda}_v = -2 \frac{v}{r} \lambda_u + \frac{u}{r} \lambda_v = f^{(6)} \quad (14)$$

where

$$a(t) = \frac{T}{m_0 + \dot{m}t} \quad (15)$$

and the boundary conditions are

$$\begin{aligned} t &= 0 & t &= t_f \text{ (unspecified)} \\ r(0) &= r_0 & r(t_f) &= r_f \\ u(0) &= u_0 & u(t_f) &= u_f \\ v(0) &= v_0 & v(t_f) &= v_f \end{aligned} \quad (16)$$

Equations (9) through (14) may be written as

$$\dot{X} = F(X, t) \quad (17)$$



where

$$\begin{aligned} X &= (x^{(1)}, \dots, x^{(6)}) \\ F &= (f^{(1)}, \dots, f^{(6)}) \end{aligned} \quad (18)$$

and

$$\begin{aligned} x^{(1)}(t) &= r(t) \quad , \quad x^{(2)}(t) = u(t) \quad , \quad x^{(3)}(t) = v(t) \\ x^{(4)}(t) &= \lambda_r(t) \quad , \quad x^{(5)}(t) = \lambda_u(t) \quad , \quad x^{(6)}(t) = \lambda_v(t) \end{aligned}$$

The method proceeds by solving the following sequence of linear two-point problems.

$$\dot{X}_{n+1} = J(X_n, t) [X_{n+1} - X_n] + F(X_n, t) \quad n = 0, 1, \dots (19)$$

where  $J(X, t)$  is the Jacobian matrix of partial derivatives of the  $f^{(i)}$  with respect to the  $x^{(j)}$ ,  $i = 1, \dots, 6$ ,  $j = 1, \dots, 6$ . A starting vector,  $X_0(t)$  and an estimated final time,  $t_{f0}$ , are assumed and the sequence of linear boundary value problems is solved numerically by the procedure outlined in (3). Altering the original boundary conditions so that the final  $r$  is maximized at  $t_{f_k}$  ( $k=0$ ) we have:

$$\begin{aligned} t = 0 & & t = t_{f_k} \\ x_n^{(1)}(0) = r_n(0) = r_0 & & \\ x_n^{(2)}(0) = u_n(0) = u_0 & & x_n^{(2)}(t_f) = u_n(t_f) = u_f \\ x_n^{(3)}(0) = v_n(0) = v_0 & & x_n^{(3)}(t_f) = v_n(t_f) = v_f \\ x_n^{(4)}(0) = \lambda_{r_n}(0) = 1 & & \\ & & n = 1, 2, \dots \end{aligned} \quad (20)$$

# COMPUTING METHODS IN OPTIMIZATION PROBLEMS

Setting  $\lambda_r(0) = 1$  accomplished the scaling of the multipliers. The iteration proceeds until  $\bar{\rho}(X_{n+1}, X_n) \leq \epsilon$  where

$$\bar{\rho}(X_{n+1}, X_n) = \sum_{i=1}^6 \max_{t \in [0, t_{f_k}]} |x_{n+1}^{(i)} - x_n^{(i)}| \quad (21)$$

At this stage the final time,  $t_{f_k}$ , is adjusted automatically according to the difference  $|r_f - r(t_{f_k})|$  by a scalar application of the Newton-Raphson procedure:

$$t_{f_{k+1}} = t_{f_k} + \frac{(t_{f_k} - t_{f_{k-1}})}{r(t_{f_k}) - r(t_{f_{k-1}})} [r_f - r(t_{f_k})] \quad (22)$$

The above iteration on  $X_n$  now continues for the new final time  $t_{f_{k+1}}$  until  $\bar{\rho}$  is again  $\leq \epsilon$ . The over-all process proceeds until  $\rho \leq \epsilon$  where

$$\rho = \bar{\rho} + \frac{1}{b} |t_{f_{k+1}} - t_{f_k}| \quad (23)$$

and  $b$  is a scaling factor. The corresponding iterate  $X_{n+1}$  is accepted as the solution to the minimum time problem, and a final check is run by integrating the non-linear Euler-Lagrange equations with a complete set of initial conditions taken from the final iterate.

For the purpose of numerical precision the data for the sample problem were normalized to obtain

$$\begin{aligned} r_0 &= 1.000 & v_f &= .8098 \\ r_f &= 1.525 & u_f &= 0.000 \\ \mu &= 1.000 & m_0 &= 1.000 \\ v_0 &= 1.000 & \dot{m} &= -.07487 \\ u_0 &= 0.000 & T &= .1405 \end{aligned} \quad (24)$$

# COMPUTING METHODS IN OPTIMIZATION PROBLEMS

This resulted in a time unit of 58.18 days. The starting vector  $X_0(t)$  was chosen rather crudely as follows:

$$t_{f_0} = 178.0 \text{ days, or } 3.060 \text{ of our time units}$$

$$x_0^{(1)}(t) = r_0(t) = r_0 + \frac{r_f - r_0}{t_{f_0}} t$$

$$x_0^{(2)}(t) = u_0(t) \equiv 0$$

$$x_0^{(3)}(t) = v_0(t) = \left( \frac{\mu}{r_0(t)} \right)^{\frac{1}{2}}$$

(25)

$$x_0^{(4)}(t) = \lambda_{r_0}(t) \equiv 1.000$$

$$x_0^{(5)}(t) = \lambda_{u_0}(t) \equiv \begin{cases} .5200 & \text{for } t \in [0, \frac{1}{2} t_{f_0}] \\ -.5000 & \text{for } t \in [\frac{1}{2} t_{f_0}, t_{f_0}] \end{cases}$$

$$x_0^{(6)}(t) = \lambda_{v_0}(t) \equiv \begin{cases} .3000 & \text{for } t \in [0, \frac{1}{2} t_{f_0}] \\ 0.000 & \text{for } t \in [\frac{1}{2} t_{f_0}, t_{f_0}] \end{cases}$$

The final two starting functions  $\lambda_{u_0}(t)$  and  $\lambda_{v_0}(t)$  correspond to a control angle  $\vartheta_0(t)$  which is constant at  $60^\circ$  above the local horizontal for the first half of the transit time, and constant inward along the local vertical for the remaining half of the transit time (see Fig. 3).

The sequence  $\{X_n\}$  converged uniformly to an accuracy of 5 significant figures in 13 total iterations with 4 shifts of the final time. The resultant minimum time was found to be 193.2 days; in agreement with the results of the gradient and second variation solutions. The total computer time (IBM 7094) required was 36 seconds. Figure 3 illustrates the behavior of the control angle program, where  $\vartheta_0(t)$  is the starting function,

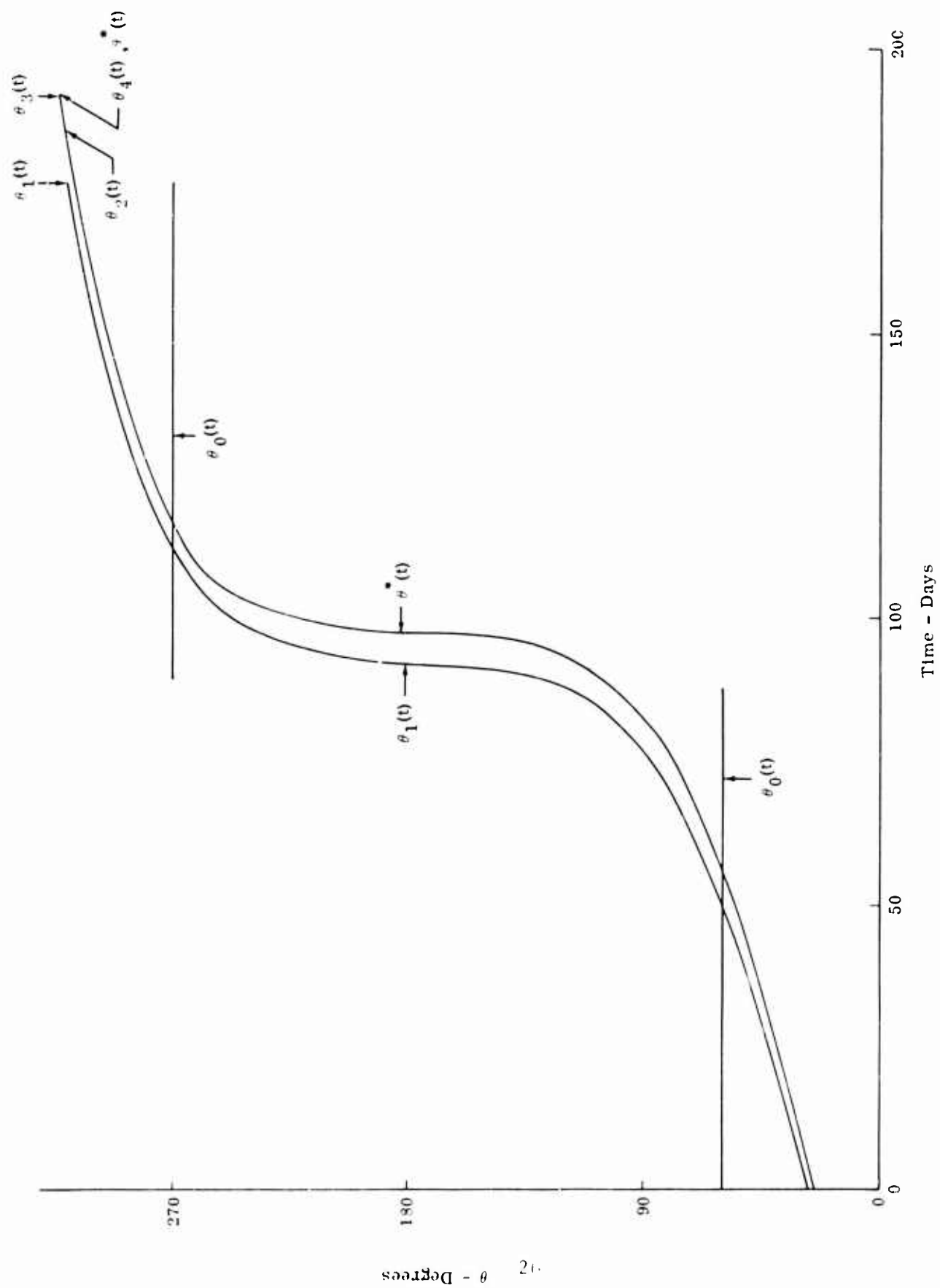


Fig. 3 Control Angle Programs from Generalized Newton-Raphson Method

## COMPUTING METHODS IN OPTIMIZATION PROBLEMS

$\phi_1(t)$  through  $\phi_4(t)$  correspond to the 4 shifts of the final time  $t_f$ , and  $\phi^*(t)$  results from the integration of the nonlinear Euler-Lagrange equations with the initial values taken from the final iterate. The curves for  $\phi_2(t)$ ,  $\phi_3(t)$ , and  $\phi_4(t)$  lie, within our plotting accuracy, on the solution curve  $\phi^*(t)$ ; except for the final segments as indicated on the figure. Figure 4 illustrates the behavior of the metrics,  $\rho$ , and  $\bar{r}$ .

We observe that for this particular example the approach just described is systematic, simple to apply, and yields rapid convergence from crude a priori starting functions.<sup>1</sup> However, it is possible that for other variational problems a priori starting functions sufficient to produce convergence may not be easily obtainable. In such a case one might consider the possibility of using a hybrid approach, that is, a combination of one of the gradient procedures, or a crude application of dynamic programming; and the technique just described. Such a hybrid approach is currently under study.

### Acknowledgments

The work reported in this paper was partly supported by the USAF Office of Scientific Research, Applied Mathematics Division, under Contract AF 49(638)-1207 and by NASA Marshall Space Flight Center, Aero-Astroynamics Laboratory, under Contract NAS 8-1549.

The authors wish to express their appreciation to Dr. Henry J. Kelley for valuable discussions and suggestions concerning this work; and to Gerald E. Taylor whose able programming made the numerical example for the generalized Newton-Raphson method possible.

### References

- (1). Lindorfer, W., and Moyer, H.G., "Application of a Low Thrust Trajectory Optimization Scheme to Planar Earth-Mars Transfer," ARS Journal, February 1962.

---

<sup>1</sup>For additional numerical examples see (7).

# COMPUTING METHODS IN OPTIMIZATION PROBLEMS

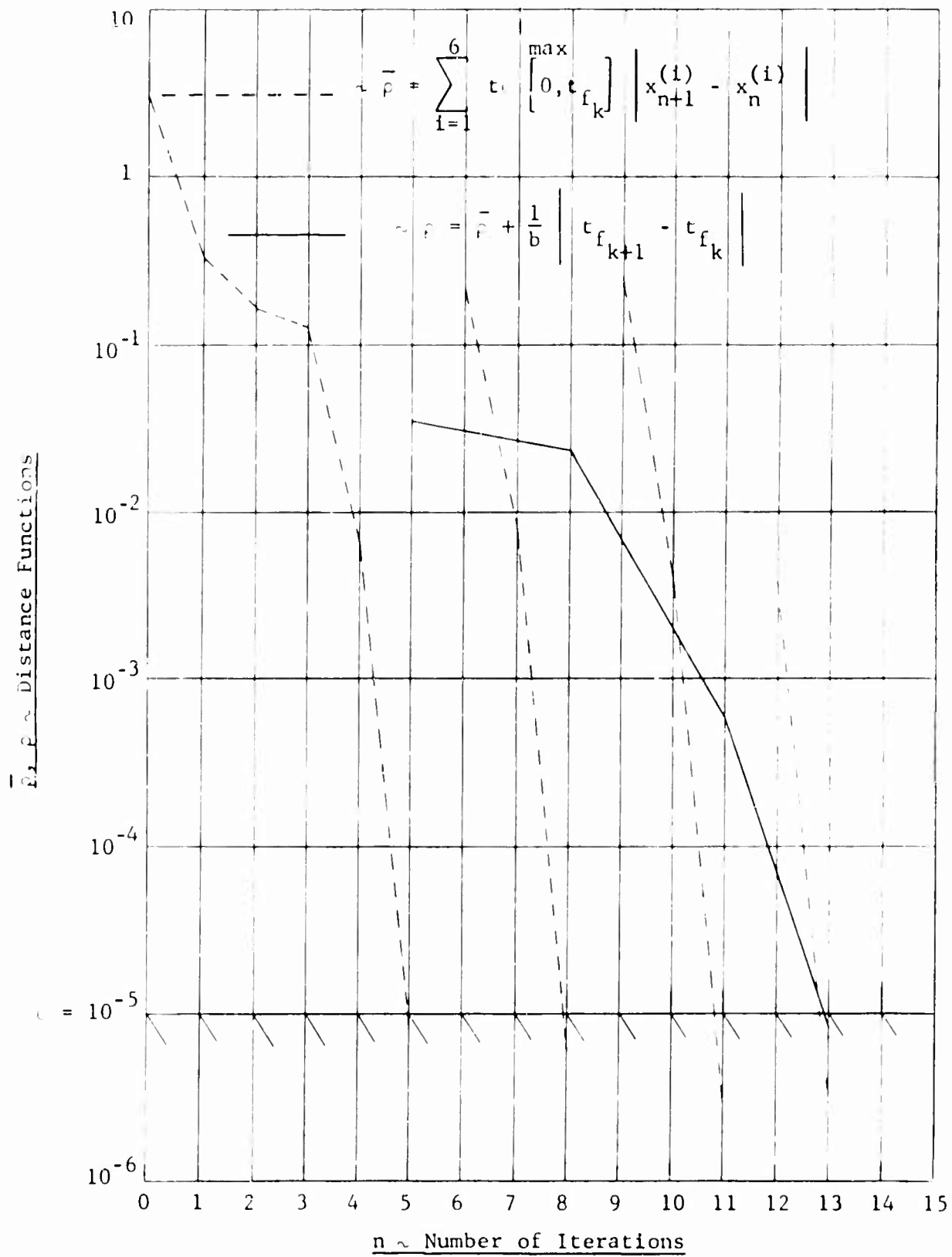


Fig. 4  $\bar{F}, F$  Versus  $n$  for the Newton-Raphson Method

# COMPUTING METHODS IN OPTIMIZATION PROBLEMS

- (2). Kelley, H.J., Kopp, R.E., and Moyer, H.G., "A Trajectory Optimization Technique Based upon the Theory of the Second Variation," presented at the AIAA Astrodynamics Conference, Yale University, Connecticut, August 1963.
- (3). McGill, R., and Kenneth, P., "A Convergence Theorem on the Iterative Solution of Nonlinear Two-Point Boundary Value Systems," presented at the XIVth IAF Congress, Paris, France, September 1963.
- (4). Kelley, H.J., "Method of Gradients," Chapter 6 of Optimization Techniques, edited by G. Leitmann, Academic Press, 1962.
- (5). Hestenes, M.R., Numerical Methods of Obtaining Solutions of Fixed End Point Problems in the Calculus of Variations, RM-102, The Rand Corporation, August 1949.
- (6). Kalaba, R., "On Nonlinear Differential Equations, the Maximum Operation, and Monotone Convergence," Journal of Mathematics and Mechanics, Vol. 8, No. 4, pp. 519-574, July 1959.
- (7). McGill, R., and Kenneth, P., "Solution of Variational Problems by Means of a Generalized Newton-Raphson Operator," to be published.